04.25-05

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant:	Jonathan S. Goldick	Examiner:	Nguyen, Cindy
Serial No.:	09/750,366	Group Art Unit:	2171
Filed:	12/27/2000	Docket:	MS155471.1/40062.95US01
Confirmation No.:	7308	Notice of Allow. Date:	n/a
Due Date:	04/21/2005		
Title:	METHOD AND SYSTEM FOR CREATING AND MAINTAINING VERSION-SPECIFIC PROPERTIES IN A FILE		

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450

27488
PATENT TRADEMARK OFFICE

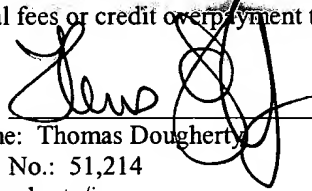Sir:

We are transmitting herewith the attached:

☒ Transmittal Sheet in duplicate containing Certificate of Mailing
☒ Check(s) in the amount of $500.00 for fee for Appeal Brief
☒ Other: Appeal Brief in triplicate
☒ Return postcard

Please consider this a PETITION FOR EXTENSION OF TIME for a sufficient number of months to enter these papers or any future reply, if appropriate. Please charge any additional fees or credit overpayment to Deposit Account No. 13-2725. A duplicate of this sheet is enclosed.

Merchant & Gould P.C.	By: _____
P.O. Box 2903	Name: Thomas Dougherty
Minneapolis, MN 55402-0903	Reg. No.: 51,214
612.332.5300	TDougherty/jw

(PTO TRANSMITTAL - GENERAL)

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant:	Jonathan S. Goldick	Examiner:	Nguyen, Cindy
Serial No.:	09/750,366	Group Art Unit:	2171
Filed:	12/27/2000	Docket No.:	MS 155471.1/40062.95US01
Title:	METHOD AND SYSTEM FOR CREATING AND MAINTAINING VERSION-SPECIFIC PROPERTIES IN A FILE		

APPELLANT'S BRIEF ON APPEAL

Mail Stop APPEAL BRIEF-PATENTS
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450

> **27488**
> PATENT TRADEMARK OFFICE

Sir:

This Brief is presented in support of the Appeal filed January 21, 2005, from the final rejection of Claims 1, 2, 5, 7-14, and 16-21 of the above-identified application, as set forth in the Office Action mailed November 23, 2004.

A check for $500.00 to cover the required fee for filing this Brief is enclosed.

Applicant reserves the right to request an oral hearing by filing a separate request for an oral hearing with the appropriate fee within two months of the date of the Examiner's Answer in response to this Brief.

Application No. 09/750,366

This brief contains these items under the following hearings, and in the order set forth below (37 CFR § 41.37(c)):

I. REAL PARTY OF INTEREST

The real party in interest in this appeal is <u>Microsoft Corporation</u>.

II. RELATED APPEALS AND INTERFERENCES

With respect to other appeals or interferences that will directly affect, or be directly affected by, or have a bearing on the Board's decision in this appeal:

☒   There are no such appeals or interferences.

☐   These are as follows:

III. STATUS OF CLAIMS

The status of the claims in this application is:

A.   TOTAL NUMBER OF CLAIMS IN APPLICATION: 21

B.   STATUS OF ALL THE CLAIMS:

    1.   Claims canceled: 3, 4, 6, and 15

    2.   Claims withdrawn from consideration but not canceled: None

    3.   Claims pending: 1, 2, 5, 7-14, and 16-21

    4.   Claims allowed: None

    5.   Claims rejected: 1, 2, 5, 7-14, and 16-21

C.   CLAIMS ON APPEAL: Claims 1, 2, 5, 7-14, and 16-21

IV. STATUS OF AMENDMENTS

A first office action was mailed on October 23, 2002, rejecting claims 1-21. A response with amendments was filed on January 22, 2003. A final office action dated March 3, 2003 maintained the rejections to claims 1-21. Applicant filed a response to the final office action on May 30, 2003, wherein claim 4 was canceled. A telephone interview was held on May 13, 2003, wherein the Examiner agreed to withdraw the finality of the previous office action. Subsequently, an office action dated June 18, 2003 rejected claims 1-3 and 5-21. Applicant filed a response with amendments to this office action on September 18, 2003. A final office action dated November 20, 2003 maintained the rejections of claims 1-3 and 5-21. Applicant filed a response to the final office action on January 20, 2004, wherein claims 3, 6 and 15 were canceled. An Advisory Action dated February 6, 2004 maintained the rejections of claims 1-3 and 5-21. Applicant filed a Request for Continued Examination on February 20, 2004. An office action dated March 26, 2004 rejected claims 1, 2, 5, 7-14, and 16-21. Applicant filed a response with amendments to this office action on June 25, 2004. A final office action dated November 23, 2004 maintained the rejection of claims 1, 2, 5, 7-14, and 16-21.

V. SUMMARY OF THE CLAIMED SUBJECT MATTER

The present invention is generally directed to the use of file attributes to provide status information to third party applications. See, e.g., Figures 2 and 3, and page 4, lines 1-7. A concise explanation of each independent claim is provided below:

A.   Independent Claim 1

Independent claim 1 generally relates to a method of providing a file stored in a computer system with version-specific information relating to a virus scanning application that is independent from an application used to create the file. See, e.g., Figure 2 and page 12, line 9-23 and page 14, line 13 - page 15, line 6. This method includes receiving a request from an application, such as a virus-scan application, to create version specific information about the virus scanning application and associating it with a file. See, e.g., Figure 2 and page 14, line 13 - page 15, line 11. The version-specific attribute contains version-specific information relating to the virus scanning application. See, e.g., Figure 2 and page 14, line 20 - page 15, line 6. The version specific information is to be maintained to reflect relevant updates to the file by automatically invalidating the version-specific information in response to a predetermined event. See, e.g., Figures 2 and 3 and page 14, line 20 - page 15, line 12. The version-specific attribute comprises mask information (314, FIG. 3) providing information related to which predetermined events invalidate the version-specific information. See, e.g., Figure 2 and 3 and page 14, line 4 - page 15, line 6. The method also includes receiving a request from the virus scanning application to evaluate the version-specific attribute. See, e.g., Figures 2 and 3 and page 14, line 20 - page 15, line 12. The method also includes providing the version specific information to the virus scanning application in response to the request to evaluate the version specific attribute. See, e.g., Figures 2 and 3 and page 14, line 22 - page 15, line 6.

B.   Independent Claim 10

Independent claim 10 generally relates to a method of accessing a file stored in a computer system, the file having an associated version-specific attribute provided by a virus

7

scanning application that is independent from an application used to create the file. See, e.g., Figure 2 and page 12, line 9-23 and page 14, line 4- 12. The version-specific attribute comprises meta information (310, FIG. 3) relating to the virus scanning application. See, e.g., Figure 3 and page 16, line 8-12. The version-specific attribute comprises mask information (314, FIG. 3) providing information related to which predetermined events invalidate the version-specific attribute. See, e.g., Figure 3 and page 16, line 19-23, and page 17, lines 10-17. The method includes receiving an access attempt relating to an access request (402, FIG. 4). See, e.g., Figure 4 and page 20, line 7-12. The method includes determining whether the access attempt relates to an invalidating access (404, FIG. 4). See, e.g., Figure 4 and page 20, line 11 - 19. The method also includes if the access attempt relates to an invalidating access, invalidating the version-specific attribute (406, FIG. 4). See, e.g., Figure 4 and page 20, line 19 - page 21, line 6. The method also includes if the access attempt relates to an invalidating access, performing the access operation related to the access request (408, FIG. 4). See, e.g., Figure 4 and page 22, lines 6-10. The method also includes if the access attempt does not relate to an invalidating access (412, FIG. 4), then performing the access operation related to the access request (408, FIG. 4). See, e.g., Figure 4 and page 22, line 11 - page 23, line 3.

C.   Independent Claim 13

Independent claim 13 generally relates to computer-readable medium having stored thereon a data structure. See, e.g., Figure 2 and 3 and page 8, line 14 - page 9, line 5. The data structure comprises an actual file data section containing actual file data. See, e.g., Figures 2 and 3 and page 16, lines 1-7. The data structure also includes a header section (212, 214, 216, 218 FIG. 2, and 302, FIG. 3). See, e.g., Figures 2 and 3 and page 12, lines 9-14, and page 16, lines 2-7. The data structure also includes a version-specific attribute section (220, 222, FIG. 2, and 308 FIG. 3), where the version-specific attribute section is created by a virus scanning application that is independent from an application used to create the actual file data. See, e.g., Figure 3 and page 16, line 8-12. The version-specific attribute section includes meta information section (310, FIG. 3). See, e.g., Figure 3 and page 16, line 8-12. The version-specific attribute also includes

8

mask information (314, FIG. 3) providing information related to which predetermined events invalidate the version-specific attribute. See, e.g., Figure 3 and page 16, line 19-23, and page 17, lines 10-17. Also where the version-specific attribute is invalidated in response to one of the predetermined events. See, e.g., Figures 2 and 3 and page 14, line 20 - page 15, line 12.

D.   Independent Claim 17

Independent claim 17 generally relates to a computer program product readable by a computer and encoding instructions for executing a computer process for managing version-specific information for a file within a file system. See, e.g., Figure 2 and 3 and page 8, line 14 - page 9, line 5. The computer process includes storing version-specific information as an attribute to create a version-specific attribute for the file. See, e.g., Figures 2 and 3 and page 14, line 20 - page 15, line 12. Where the version-specific attribute comprises meta information relating to a virus scanning application that is independent from an application used to create the file. See, e.g., Figure 3 and page 16, line 8-12. Also where the version specific attribute includes mask information providing information related to which predetermined events invalidate the version-specific attribute. See, e.g., Figures 2 and 3 and page 14, line 20 - page 15, line 12. The process further includes invalidating the version-specific attribute in response to a predetermined access attempt. See, e.g., Figure 4 and page 20, line 11 - page 21, line 6.

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Claims 1, 2, 5, 7-14, and 16-21 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Hodges et al. (USPN 6,035,423) in view of Nachenberg (USPN 6,067,410).

VII. ARGUMENT

A.   REJECTION OF INDEPENDENT CLAIMS 1, 10, 13 AND 17

Prior to addressing the merits of final rejections to claims 1, 10, 13, and 17 in view of Hodges and Nachenberg, it is helpful to briefly describe the teachings of Hodges and Nachenberg that the Examiner is relying on as the basis for maintaining these rejections.

Hodges et al.

In general Hodges describes a system or method having the specific problems that are solved by the present invention. As stated in Hodges' Invention Summary (Column 4, lines 54-57), "The central antivirus server comprises a first database containing information related to the latest antivirus software updates contained on each local computer." Whereas Hodges teaches said central database of version information, the present invention uses an entirely different, distributed version data storage model. Specifically, the present invention can store version information in every virus-scanned file on the user's computer, via the use of adding or modifying file attributes, instead of relying on a single, centralized database as in Hodges.

Hodges is directed to a method for updating antivirus files on a computer using "push" technology. More specifically, the files of a single anti-virus package are periodically checked for a newer version, and automatically updated or replaced if one exists via a push method. Said checking can occur over a local area network to a central server, or over the Internet. Hodges only relates to versions of virus definition files within the anti-virus software package itself, and does not relate to the status of particular user files, thus further distinguishing it from the present invention.

Nachenberg

Nachenberg is directed to an emulation repair system that restores virus-infected files via a virtual machine. Through emulation, the original (virus-free) state of the infected file is determined, and said infected file can be restored to this state. Nachenberg is wholly inapposite to the mask information claimed in the present invention, as such information is completely

unnecessary to its methodology. It does not teach or suggest mask information as in the present invention, nor does it teach or suggest predetermined events being used to invalidate version-specific information. The latter elements simply have no analogy in Nachenberg.

Applicant respectfully traverses the § 103(a) rejections of claims 1, 2, 5, 7-14, and 16-21, since the combination of Hodges and Nachenberg fails to teach or suggest the invention recited in each of these claims. Specifically, a prima facie case of obviousness first and foremost requires that the cited references must teach or suggest each of the elements of the claimed invention (MPEP § 706.02(j) and 2142-43). However, the combination of Hodges and Nachenberg fails to disclose or suggest the use of "a version specific attribute," or "mask information providing information related to which predetermined events invalidate the version-specific information" as recited in independent claims 1, 10, 13, and 17. Reconsideration of the § 103(a) rejections is therefore respectfully requested. Claims 2, 5, 7-9, 14, 16, and 18-21 depend on these independent claims, and thus should be allowed for at least the same reasons.

The Office Action of March 26, 2004 relied primarily on the Hodges reference in making the obviousness rejections of the independent claims, with Nachenberg being cited only for the purported disclosure of "version-specific attributes compris[ing] mask information providing information related to which predetermined events invalidate the version-specific information."

Hodges seeks to solve the problem of keeping an anti-virus software package up to date. Nachenberg seeks to solve the problem of fixing virus-infected files. The present invention seeks to solve still another problem, namely that of tracking which files have been scanned by the current version of an anti-virus software package, and which events can nullify the "has been scanned" status. The three inventions are thus directed at solving three entirely different problems.

Hodges describes a system or method having the specific problems that are solved by the present invention. As stated in Hodges' Invention Summary (Column 4, lines 54-57), "The central antivirus server comprises a first database containing information related to the latest antivirus software updates contained on each local computer." Whereas Hodges teaches said

central database of version information, the present invention uses an entirely different, distributed version data storage model. Specifically, the present invention can store version information in every virus-scanned file on the user's computer, via the use of adding or modifying file attributes, instead of relying on a single, centralized database as in Hodges.

As stated above, Hodges is directed to a method for updating antivirus files on a computer using "push" technology where the files of a single anti-virus package are periodically checked for a newer version, and automatically updated or replaced if one exists via a push method. More specifically, Hodges does <u>not</u> describe the creation or later use of a **version-specific attribute that pertains to the version of the virus-definition file and is located with the file itself.** Simply put, Hodges does not add its own version-specific attributes to user files (i.e., files made by independent applications) but rather only checks and updates the virus-package files.

Nachenberg does not satisfy the inadequacies of Hodges. Since Nachenberg is only directed to restoring corrupt files, Nachenberg does not teach or describe any of the limitations of the claimed invention. It does not teach or suggest mask information as in the present invention, nor does it teach or suggest predetermined events being used to invalidate version-specific information.

The combination of Hodges and Nachenberg simply does not teach or suggest each of the elements of the claimed invention. Neither Hodges or Nachenberg, alone or in combination, disclose the use of "a version specific attribute," or "mask information providing information related to which predetermined events invalidate the version-specific information" as recited in independent claims 1, 10, 13, and 17.

In support of the outstanding rejection to Claim 1, the Examiner contends that the motivation to combine the references is as follows, "The motivation being to enable the system comprising [sic] the emulation module to control decryption of these viruses that are encrypted (col. 2, lines 55 to col. 3, lines 11, Nachenberg." Such a conclusory statement is not a sufficient

13

suggestion or motivation to modify the teachings of Hodges and Nachenberg. Rather, the motivation for such modification must be explicit in the teachings of Hodges and Nachenberg or implicit in the prior art, neither of which has been established by the Examiner. See WMS Gaming, Inc. v. International Game Tech., 184 F.3d 1339, 1355, 51 U.S.P.Q.2d 1385, 1397 (Fed. Cir. 1999); Ex Parte Krantz, 19 U.S.P.Q.2d 1216, 1218 (Bd.Pat.App.&Interf. 1991).

The Examiner's obviousness rejections are instead impermissibly based on hindsight reconstruction derived by applying Applicants' claimed invention as a template to describe how the Hodges and Nachenberg references may be combined to describe something very dissimilar to the subject of the present disclosure. See In re Fritch, 972 F.2d 1260, 23 U.S.P.Q.2d 1780 (Fed.Cir. 1992); See In re Gorman, 933 F.2d 982, 18 U.S.P.Q.2d 1885 (Fed.Cir. 1991). The use of improper hindsight to find such application has been consistently disavowed by the courts. Although the tendency to resort to hindsight is often difficult to avoid, the Examiner's legal conclusion must be reached on the basis of the facts gleaned from the prior art. MPEP §2142. In the case at hand, the Examiner's legal conclusion is based solely on hindsight improperly gleaned from the present disclosure.

In sum, the instant rejections to independent claims 1, 10, 13, and 17 are improper due to the Examiner's failure to establish a prima facie case of obviousness with regard to any of these claims. First, Hodges and Nachenberg do not teach or suggest all the recited claim limitations. Furthermore, there is no suggestion or motivation to combine Hodges and Nachenberg in the references, or in the knowledge generally available to one of ordinary skill in the art at the time the invention was made. The Examiner is relying on hindsight reconstruction to maintain these rejections. The rejection of the independent claims 1, 10, 13, and 17 should be withdrawn and these claims allowed.

B.   REJECTION OF DEPENDENT CLAIMS 2, 5, 7-14, 16, and 18-21

Dependent claims 2, 5, 7-9, 14, 16, and 18-21 stand rejected along with independent claims 1, 10, 13, and 17 as obvious in light of Hodges and Nachenberg. For the reasons set forth above, Applicant submits that the Hodges and Nachenberg do not render Applicant's invention set forth in any of these independent claims obvious and that the Examiner has failed to establish
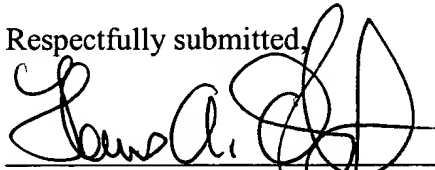
a prima facie case of obviousness with regard to same. Consequently, claims 2, 5, 7-9, 14, 16, and 18-21, which are dependent from, and thereby include each of the recitations of claim 1, 10, 13, and 17, are also believed allowable over Hodges and Nachenberg.

VIII. SUMMARY

In view of the foregoing remarks, Applicant earnestly requests that the Examiner's rejection be reversed, and that all of the pending claims be allowed. As noted above, a check in the amount of $500 is enclosed herewith to cover the fee required for submission of this Appeal Brief. It is believed that no further fees are due. However, the Commissioner is hereby authorized to charge any deficiencies or credit any overpayment to deposit account number 13-2725.

Respectfully submitted,

Dated: 4/21/05

Thomas A. Dougherty, Reg. No. 51,214
MERCHANT & GOULD P.C.
P.O. Box 2903
Minneapolis, MN 55402-0903
303.357.1642

IX. CLAIMS APPENDIX

1.   (Previously presented) A method of providing a file stored in a computer system with version-specific information relating to a virus scanning application that is independent from an application used to create the file, the method comprising:

receiving a request from the virus scanning application to create a version-specific attribute associated with the file, wherein the version-specific attribute contains version-specific information relating to the virus scanning application;

maintaining the version-specific attribute to reflect relevant updates to the file by automatically invalidating the version-specific information in response to a predetermined event, wherein the version-specific attribute comprises mask information providing information related to which predetermined events invalidate the version-specific information;

receiving a request from the virus scanning application to evaluate the version-specific attribute; and

providing the version specific information to the virus scanning application in response to the request to evaluate the version specific attribute.

2.   (Previously presented) A method as defined in claim 1 wherein the version-specific information relates to a version of a virus definition file.

5.   (Original) A method as defined in claim 1 wherein the predetermined event is an update to file data.

7.   (Original) A method of as defined in claim 1 wherein the method further comprises:

providing security information within the version-specific attribute.

8.   (Original) A method as defined in claim 1 wherein the invalidating act further comprises deleting the version-specific attribute.

9.   (Original) A computer program product readable by a computer and encoding instructions for executing the method recited in claim 1.

17

10.   (Previously presented) A method of accessing a file stored in a computer system, the file having an associated version-specific attribute provided by a virus scanning application that is independent from an application used to create the file, wherein the version-specific attribute comprises meta information relating to the virus scanning application and mask information providing information related to which predetermined events invalidate the version-specific attribute, said method comprising:

        receiving an access attempt relating to an access request;

        determining whether the access attempt relates to an invalidating access;

        if the access attempt relates to an invalidating access:

            invalidating the version-specific attribute; and

            performing the access operation related to the access request; and

        if the access attempt does not relate to an invalidating access then performing the access operation related to the access request.

11.   (Previously presented) A method of accessing a file as defined in claim 10 wherein if the access attempt does not relate to an invalidating access, then said method further comprises:

        determining whether the access depends on the version specific attribute;

        if the access does not depend on the version specific attribute, then performing the access operation related to the access request; and

        if the access depends on the version specific attribute:

        determining whether the attribute is valid; and

        performing a predetermined operation with the virus scanning application based on whether the attribute is determined to be valid.

12.   (Original) A computer program product readable by a computer and encoding instructions for executing the method recited in claim 10.

18

13. (Previously presented) A computer-readable medium having stored thereon a data structure, wherein the data structure comprises:

an actual file data section containing actual file data;

a header section; and

a version-specific attribute section, wherein the version-specific attribute section is created by a virus scanning application that is independent from an application used to create the actual file data, and wherein the version-specific attribute comprises a meta information section and a mask information section, the mask information section providing information related to which predetermined events invalidate the version-specific attribute, and wherein the version-specific attribute is invalidated in response to one of said predetermined events.

14. (Previously presented) A computer-readable medium as defined in claim 13 wherein the meta information section stores the name of the attribute, and wherein the version-specific attribute further comprises:

a version information section for storing information related to the version of a virus definition file used with the virus scanning.

16. (Previously presented) A computer readable medium as defined in claim 14 wherein the predetermined event relates to a modification of the data structure.

17. (Previously presented) A computer program product readable by a computer and encoding instructions for executing a computer process for managing version-specific information for a file within a file system, said computer process comprising:

storing version-specific information as an attribute to create a version-specific attribute for the file, wherein the version-specific attribute comprises meta information relating to a virus scanning application that is independent from an application used to create the file and mask information providing information related to which predetermined events invalidate the version-specific attribute; and

invalidating the version-specific attribute in response to a predetermined access attempt.

18.   (Previously presented) A computer program product as defined in claim 17 wherein the computer comprises a file system and the acts of storing and invalidating are performed by the file system.

19.   (Previously presented) A computer program product as defined in claim 17 wherein the version-specific information relates to a virus definition file used with the virus scanning application.

20.   (Previously presented) A computer program product as defined in claim 19 wherein the predetermined access attempt relates to a modification of the file.

21.   (Original) A computer program product as defined in claim 20 wherein the version-specific attribute remains following one of the following access attempts: copy, rename or backup.

X. **EVIDENCE APPENDIX**

None

XI. RELATED PROCEEDINGS APPENDIX

None